Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'systemd-dissect.1' command

$ man systemd-dissect.1

SYSTEMD-DISSECT(1) systemd-dissect SYSTEMD-DISSECT(1)

NAME

systemd-dissect - Dissect Discoverable Disk Images (DDIs)

SYNOPSIS

systemd-dissect [OPTIONS...] IMAGE

systemd-dissect [OPTIONS...] --mount IMAGE PATH

systemd-dissect [OPTIONS...] --umount PATH

systemd-dissect [OPTIONS...] --copy-from IMAGE PATH [TARGET]

systemd-dissect [OPTIONS...] --copy-to IMAGE [SOURCE] PATH

DESCRIPTION

systemd-dissect is a tool for introspecting and interacting with file system OS disk images, specifically Discoverable Disk Images (DDIs). It supports five different operations:

1. Show general OS image information, including the image's os-release(5) data, machine ID, partition information and more.
2. Mount an OS image to a local directory. In this mode it will dissect the OS image and mount the included partitions according to their designation onto a directory and possibly sub-directories.
3. Unmount an OS image from a local directory. In this mode it will recursively unmount the mounted partitions and remove the underlying loop device, including all the partition sub-devices.
4. Copy files and directories in and out of an OS image.

The tool may operate on three types of OS images:

1. OS disk images containing a GPT partition table envelope, with partitions marked according to the Discoverable Partitions Specification[1].
2. OS disk images containing just a plain file-system without an enveloping partition table. (This file system is assumed to be the root file system of the OS.)
3. OS disk images containing a GPT or MBR partition table, with a single partition only. (This partition is assumed to contain the root file system of the OS.)

OS images may use any kind of Linux-supported file systems. In addition they may make use of LUKS disk encryption, and contain Verity integrity information. Note that qualifying OS images may be booted with systemd-nspawn(1)'s --image= switch, and be used as root file system for system service using the RootImage= unit file setting, see systemd.exec(5).

Note that the partition table shown when invoked without command switch (as listed below) does not necessarily show all partitions included in the image, but just the partitions that are understood and considered part of an OS disk image. Specifically, partitions of unknown types are ignored, as well as duplicate partitions (i.e. more than one per partition type), as are root and /usr/ partitions of architectures not compatible with the local system. In other words: this tool will display what it operates with when mounting the image. To display the complete list of partitions use a tool such as fdisk(8).

COMMANDS

If neither of the command switches listed below are passed the specified disk image is opened and general information about the image and the contained partitions and their use is shown.

--mount, -m

Mount the specified OS image to the specified directory. This will dissect the image, determine the OS root file system ? as well as possibly other partitions ? and mount them to the specified directory. If the OS image contains multiple partitions marked with the Discoverable Partitions Specification[1] multiple nested mounts

are established. This command expects two arguments: a path to an image file and a path to a directory where to mount the image.

To unmount an OS image mounted like this use the --umount operation.

When the OS image contains LUKS encrypted or Verity integrity protected file systems appropriate volumes are automatically set up and marked for automatic disassembly when the image is unmounted.

The OS image may either be specified as path to an OS image stored in a regular file or may refer to block device node (in the latter case the block device must be the "whole" device, i.e. not a partition device). (The other supported commands described here support this, too.)

All mounted file systems are checked with the appropriate fsck(8) implementation in automatic fixing mode, unless explicitly turned off (--fsck=no) or read-only operation is requested (--read-only).

-M

This is a shortcut for --mount --mkdir.

--umount, -u

Unmount an OS image from the specified directory. This command expects one argument: a directory where an OS image was mounted. All mounted partitions will be recursively unmounted, and the underlying loop device will be removed, along with all it's partition sub-devices.

-U

This is a shortcut for --umount --rmdir.

--copy-from, -x

Copies a file or directory from the specified OS image into the specified location on the host file system. Expects three arguments: a path to an image file, a source path (relative to the image's root directory) and a destination path (relative to the current working directory, or an absolute path, both outside of the image). If the destination path is omitted or specified as dash ("-"), the specified file is written to standard output. If the

source path in the image file system refers to a regular file it is copied to the destination path. In this case access mode, extended attributes and timestamps are copied as well, but file ownership is not. If the source path in the image refers to a directory, it is copied to the destination path, recursively with all containing files and directories. In this case the file ownership is copied too.

--copy-to, -a

Copies a file or directory from the specified location in the host file system into the specified OS image. Expects three arguments: a path to an image file, a source path (relative to the current working directory, or an absolute path, both outside of the image) and a destination path (relative to the image's root directory). If the source path is omitted or specified as dash ("-"), the data to write is read from standard input. If the source path in the host file system refers to a regular file, it is copied to the destination path. In this case access mode, extended attributes and timestamps are copied as well, but file ownership is not. If the source path in the host file system refers to a directory it is copied to the destination path, recursively with all containing files and directories. In this case the file ownership is copied too.

As with --mount file system checks are implicitly run before the copy operation begins.

-h, --help

Print a short help text and exit.

--version

Print a short version string and exit.

OPTIONS

The following options are understood:

--read-only, -r

Operate in read-only mode. By default --mount will establish writable mount points. If this option is specified they are

established in read-only mode instead.

--fsck=no

Turn off automatic file system checking. By default when an image is accessed for writing (by --mount or --copy-to) the file systems contained in the OS image are automatically checked using the appropriate fsck(8) command, in automatic fixing mode. This behavior may be switched off using --fsck=no.

--growfs=no

Turn off automatic growing of accessed file systems to their partition size, if marked for that in the GPT partition table. By default when an image is accessed for writing (by --mount or --copy-to) the file systems contained in the OS image are automatically grown to their partition sizes, if bit 59 in the GPT partition flags is set for partition types that are defined by the Discoverable Partitions Specification[1]. This behavior may be switched off using --growfs=no. File systems are grown automatically on access if all of the following conditions are met:

1. The file system is mounted writable
2. The file system currently is smaller than the partition it is contained in (and thus can be grown)
3. The image contains a GPT partition table
4. The file system is stored on a partition defined by the Discoverable Partitions Specification
5. Bit 59 of the GPT partition flags for this partition is set, as per specification
6. The --growfs=no option is not passed.

--mkdir

If combined with --mount the directory to mount the OS image to is created if it is missing. Note that the directory is not automatically removed when the disk image is unmounted again.

--rmdir

If combined with --umount the specified directory where the OS image is mounted is removed after unmounting the OS image.

--discard=

Takes one of "disabled", "loop", "all", "crypto". If "disabled" the image is accessed with empty block discarding turned off. If "loop" discarding is enabled if operating on a regular file. If "crypt" discarding is enabled even on encrypted file systems. If "all" discarding is unconditionally enabled.

--root-hash=, --root-hash-sig=, --verity-data=

Configure various aspects of Verity data integrity for the OS image. Option --root-hash= specifies a hex-encoded top-level Verity hash to use for setting up the Verity integrity protection. Option --root-hash-sig= specifies the path to a file containing a PKCS#7 signature for the hash. This signature is passed to the kernel during activation, which will match it against signature keys available in the kernel keyring. Option --verity-data= specifies a path to a file with the Verity data to use for the OS image, in case it is stored in a detached file. It is recommended to embed the Verity data directly in the image, using the Verity mechanisms in the Discoverable Partitions Specification[1].

--no-pager

Do not pipe output into a pager.

--no-legend

Do not print the legend, i.e. column headers and the footer with hints.

--json=MODE

Shows output formatted as JSON. Expects one of "short" (for the shortest possible output without any redundant whitespace or line breaks), "pretty" (for a pretty version of the same, with indentation and line breaks) or "off" (to turn off JSON output, the default).

EXIT STATUS

On success, 0 is returned, a non-zero failure code otherwise.

SEE ALSO

systemd(1), systemd-nspawn(1), systemd.exec(5), Discoverable Partitions

Specification[1], umount(8), fdisk(8)

NOTES

1. Discoverable Partitions Specification

https://systemd.io/DISCOVERABLE_PARTITIONS